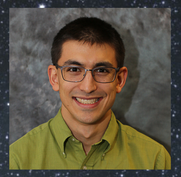


aj.riggs@jpl.nasa.gov

github.com/ajeldorado/falco-matlab

Fast Linearized Coronagraph Optimizer (FALCO)

A.J. Eldorado Riggs*, Garreth Ruane†, Erkin Sidick*, Carl Coker*, Brian Kern*

* Jet Propulsion Laboratory, California Institute of Technology
† California Institute of Technology

Introduction

FALCO is an **open-source, modular software** collection for **coronagraphy** and wavefront sensing and control (**WFSC**).

1. Provides example WFSC code and models for several coronagraph types.
   a) Can be used for
      i. coronagraph **design**
      ii. optical **modeling** (of WFSC, tolerances, etc)
      iii. high-level **testbed operation**
   b) Builds upon the **PROPER** optical propagation library by J. Krist.

2. Provides a *community software platform* for researchers to:
   a) save hundreds of hours from not having to re-invent coronagraph design/modeling/testing code from scratch.
   b) See and run examples of "standard" WFSC algorithms for coronagraphy (e.g., Speckle Nulling and EFC).
   c) Contribute working examples of new algorithms by writing new functions for FALCO.

Availability: Open-Source at Github.com

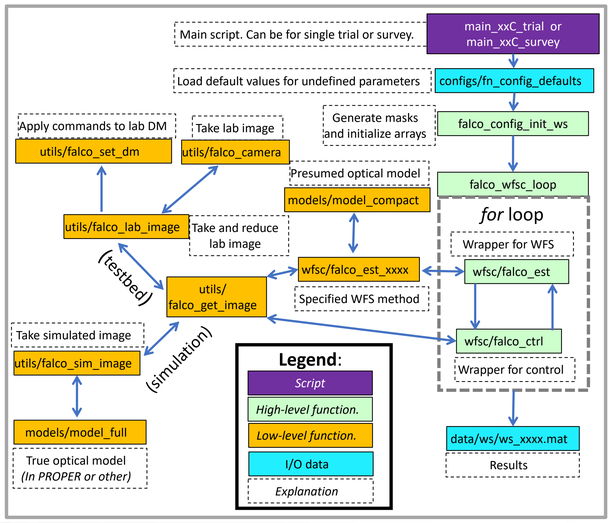
- **MATLAB**: **github.com/ajeldorado/falco-matlab**
- **Python 3.6**: (*available in late 2018*)

Key Features of FALCO

1. **Rapidly calculates DM-response matrices** (aka control Jacobians)
   a) Algorithms used are *orders of magnitude faster* than conventional, direct methods.
   b) Achieved using **matrix Fourier transforms (MFTs)** and **sub-array propagation** as appropriate.
   c) May enable on-orbit calculations of control Jacobians.
2. **Testbed-specific code is modular and portable** to different high-contrast coronagraph testbeds.
3. Provides **many example scripts** to use as starting points.

Code Structure

- FALCO (in Matlab) has a versatile, modular code structure with a straightforward data flow.
  - Easily toggle between testbed and simulation
- Python version will be re-designed to be object-oriented.



Current Features

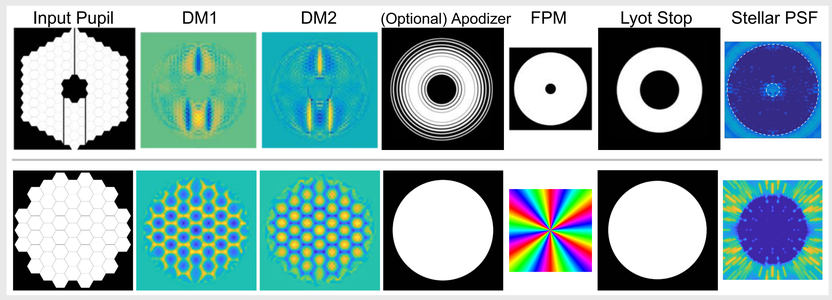
- Support for many coronagraphs (Lyot, hybrid Lyot, vortex, APP, SPC, APLC)
- Electric Field Conjugation (EFC)
- Xinetics DM models
- Parallelization of large calculations.
- Functions to generate pupil masks: LUVOIR A/B, WFIRST, HabEx, Keck

Upcoming Features

- Pair-wise probing estimator (as batch process and Kalman filter) Speckle Nulling controller
- Testbed compatibility
  - Being tested on Caltech's HCST
  - Earlier versions of FALCO have run the testbeds at Princeton, JPL, and STScI
- Calibration routines:
  - Phase retrieval
  - Image sharpening
- Boston Micromachines DM model

Examples

Examples of FALCO for DM-integrated designs with two mission concepts.



Related Posters & Talks

- FALCO II: Validation, E. Sidick, Paper 10698-165
- FALCO III: Trade Studies, C. Coker, Paper 10698-174
- FALCO IV: Design Survey, G. Ruane, Paper 10698-167
- HCST Results, G. Ruane, Paper 10698-50

References

- J. Krist, "PROPER", Proc. of SPIE Vol. 6675, 66750P. 2007.
- Soummer et al., "Fast Computation", Optics Express Vol. 15, 24. 2007.

FALCO's development was funded in part by NASA Exoplanet Exploration Program's (**ExEP**) Segmented Coronagraph Design and Analysis (**SCDA**) study.
This work was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.